



**FORMULACIÓN DE UN MODELO MATEMÁTICO PARA OPTIMIZAR LA
PROGRAMACIÓN DE VISITAS DE CONTROL A PRIVADOS DE LA LIBERTAD
CON MEDIDA DOMICILIARIA USANDO UN ALGORITMO GENÉTICO
MODIFICADO PARA EL ESTABLECIMIENTO PENITENCIARIO DE MEDIANA
SEGURIDAD Y CARCELARIO DE PEREIRA**

TRABAJO DE GRADO COMO REQUISITO PARA OPTAR AL TÍTULO DE
MAGÍSTER EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

FACULTAD DE INGENIERÍAS
POSTGRADOS

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN



**FORMULACIÓN DE UN MODELO MATEMÁTICO PARA OPTIMIZAR LA
PROGRAMACIÓN DE VISITAS DE CONTROL A PRIVADOS DE LA LIBERTAD
CON MEDIDA DOMICILIARIA USANDO UN ALGORITMO GENÉTICO
MODIFICADO PARA EL ESTABLECIMIENTO PENITENCIARIO DE MEDIANA
SEGURIDAD Y CARCELARIO DE PEREIRA**

Anderson Danilo Betancourt Betancourt
Juan Carlos Gutiérrez Martínez

Asesor
PhD. Jorge Iván Ríos Patiño

FACULTAD DE INGENIERÍAS

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

NOTA DE ACEPTACIÓN:

Nombre director, orientador, asesor

Firma jurado (Nombres)

Firma Jurado (Nombres)

Pereira - Risaralda, Junio 2018

Dedicamos esta tesis a Dios, por darnos la fuerza para llegar a la meta a pesar de las dificultades del camino.

A nuestras Madres que siempre creyeron en nosotros sin perder la fe.

AGRADECIMIENTO

Los autores expresan sus agradecimientos a:

A nuestro director de tesis PhD Jorge Iván Ríos; gracias por brindarnos una amistad, por animarnos a no desfallecer en este difícil camino, por creer fielmente en el proyecto y por sus aportes profesionales para lograr el objetivo.

Agradecemos a Jonathan Gutiérrez, miembro del grupo de Investigación Sirius de la Universidad Tecnológica de Pereira; quien con sus conocimientos en Optimización y en el lenguaje de programación Python hizo importantes aportes a la construcción del algoritmo que permitió probar el modelo matemático propuesto por los autores.

TABLA DE CONTENIDO

| | Pág. |
|--|------|
| INTRODUCCIÓN..... | 15 |
| 1. TITULO..... | 17 |
| 2. DEFINICIÓN DEL PROBLEMA..... | 18 |
| 3. JUSTIFICACIÓN..... | 19 |
| 4. OBJETIVOS..... | 20 |
| 4.1 OBJETIVO GENERAL..... | 20 |
| 4.2 OBJETIVOS ESPECÍFICOS..... | 20 |
| 5. MARCO REFERENCIAL..... | 21 |
| 5.1 MARCO TEÓRICO..... | 21 |
| 5.1.1 Inteligencia artificial..... | 21 |
| 5.2 MARCO CONCEPTUAL..... | 22 |
| 5.2.1 Algoritmos genéticos..... | 22 |
| 5.2.2 Heurística..... | 22 |
| 5.2.3 Metaheurística..... | 23 |
| 6. DESAROLLO METODOLÓGICO..... | 25 |
| 7. CAPÍTULO I..... | 26 |
| 7.1 EXPERTOS PENITENCIARIOS:..... | 27 |
| 8. CAPÍTULO II..... | 30 |
| 8.1 NOMENCLATURA..... | 30 |
| 8.2 ANÁLISIS DEL ESFUERZO COMPUTACIONAL..... | 33 |
| 8.3 MODELO MATEMÁTICO..... | 34 |
| 8.4 HEURÍSTICAS PARA LA EJECUCIÓN DEL ALGORITMO..... | 35 |
| 8.5 CODIFICACIÓN DEL PROBLEMA..... | 35 |
| 9. ALGORÍTMO..... | 37 |
| 9.1 PSEUDOCÓDIGO..... | 37 |

| | | |
|-------|---------------------------------------|----|
| 9.2 | PYTHON..... | 37 |
| 9.3 | OPERACIÓN DEL ALGORITMO GENÉTICO..... | 38 |
| 9.3.1 | Tamaño de la Población | 38 |
| 9.3.2 | Codificación de las Variables | 38 |
| 9.3.3 | Codificación Numérica..... | 38 |
| 9.3.4 | Selección por Rueda de Ruleta | 39 |
| 9.3.5 | Reproducción o Crossover..... | 39 |
| 9.3.6 | Mutación..... | 40 |
| 9.3.7 | Reemplazo | 41 |
| 9.3.8 | Criterio de parada..... | 41 |
| 9.4 | RESULTADOS OBTENIDOS..... | 41 |
| 9.4.1 | Parámetros de ejecución | 44 |
| 9.4.2 | Graficas..... | 45 |
| 10. | CONCLUSIONES Y TRABAJOS FUTUROS | 47 |
| 11. | BIBLIOGRAFÍA | 48 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1 Variables del Modelo Matemático..... | 30 |
| Tabla 2 Resultados de pruebas de ejecución | 41 |
| Tabla 3 Parametros de ejecución para el algoritmo | 44 |

ÍNDICE DE GRÁFICAS

| | |
|--|----|
| Gráfica 1 Longitud del Individuo Vs Visitados | 45 |
| Gráfica 2 Función objetivo vs Longitud del Individuo | 46 |

ÍNDICEDE ILUSTRACIONES

| | |
|---|----|
| Ilustración 1. Aplicativo Sisipec Web INPEC..... | 28 |
| Ilustración 2. Prototipo sistema DAGA - Mapa de PPL y PNVCC..... | 29 |
| Ilustración 4. Ejemplo Método (PMX) Partially Mapped Crossover..... | 40 |
| Ilustración 5. Mutación..... | 41 |

ÍNDICE DE ANEXOS

Anexo A ... Código fuente de Algoritmo Genético en lenguaje de programación Python

Anexo B ... Archivo *data.json* el cual contiene la información asociada a los PPL

Anexo C... Entrevista al experto Penitenciario MICHAEL MAHECHA RÚA, formato mp3

RESUMEN

En Colombia más de 60.000 personas privadas de la libertad se encuentran en prisión y detención domiciliaria¹, de los cuales más de 1.000 pertenecen al establecimiento carcelario de Pereira, situación que amenaza la seguridad ciudadana. Actualmente, las autoridades penitenciarias y de policía encargadas del control de la pena impuesta por la autoridad judicial planean las visitas de control al cumplimiento de la medida de forma manual, lo que hace que el proceso sea lento y que el servidor penitenciario que haga la planeación de la ruta pueda estar sesgado al momento de seleccionar los individuos a ser visitados.

Este proyecto presenta un modelo matemático diseñado para optimizar la planeación de visitas de control a personas privadas de la libertad con medida de aseguramiento en el domicilio, y haciendo uso de la inteligencia artificial ofrece un algoritmo genético modificado para la solución del modelo de optimización propuesto.

Se muestra como los algoritmos genéticos están desempeñando un papel cada vez más importante en diferentes áreas de la ciencia y ahora ayudan a optimizar tareas del sistema penitenciario y el cumplimiento de la ley penal.

Comentado [JCGM1]: DECIDIR: ¿Lo retiramos?

Palabras clave: Inteligencia Artificial, Heurísticas, Metaheurísticas, Algoritmos Genéticos, Optimización, Prisión Domiciliaria.

¹ Tablero estadístico INPEC, consultado el 20-05-2018 http://201.217.206.18:8080/jasperserver-pro/flow.html?_flowId=dashboardRuntimeFlow&dashboardResource=/public/Domiciliarias/Dashboards/Tablero_Domiciliarias_Colombia&j_username=inpec_user&j_password=inpec

ABSTRACT

In Colombia more than 60,000 people deprived of liberty are in prison and house arrest, of which more than 1,000 belong to the prison of Pereira, a situation that threatens public safety. Currently, the prison and police authorities in charge of the control of the penalty imposed by the judicial authority plan the control visits to comply with the measure manually, which makes the process slow and the prison server who makes the planning of the route may be biased when selecting the individuals to be visited.

This project presents a mathematical model designed to optimize the planning of control visits to people deprived of freedom with an assurance measure at home and making use of artificial intelligence offers a modified genetic algorithm for the solution of the proposed optimization model.

It shows how genetic algorithms are playing an increasingly important role in different areas of science and now **help** to optimize tasks of the penitentiary system and compliance with criminal law.

Keywords: Artificial Intelligence, Heuristics, Metaheuristics Genetic Algorithms, Optimization, Domiciliary Prison.

Comentado [JCGM2]: DECIDIR: ¿Lo retiramos?

GLOSARIO

INPEC: Instituto Nacional Penitenciario y Carcelario.

SISIPECWEB: Sistematización Integral del Sistema Penitenciario y Carcelario, sistema de información donde se registra los datos de los internos que se encuentran a cargo del INPEC.

INTRODUCCIÓN

En el contexto de la cotidianidad existen problemas que, aunque pueden, en principio, parecer de sencilla solución, al tratar de llevarlos a la práctica se comienza a detectar la dificultad para resolverlos.

Un problema clásico es el denominado Problema de la mochila (*in. Knapsack problem*), en la actualidad ampliamente documentado[1], el cual, en términos generales, consiste en guardar elementos, cada uno con un volumen y un costo asociado, en una mochila con una capacidad de volumen determinada, de forma tal que la sumatoria de los costos todos los productos que son almacenados en la mochila, sea máxima.

Para pequeñas instancias del problema, en las que la sumatoria del costo de los productos almacenados en la mochila es computable en un tiempo razonable, teniendo en cuenta la capacidad de la mochila, se pueden utilizar métodos que pueden lograr la solución exacta.

En el caso planteado es necesario determinar qué es una pequeña instancia del problema y un método exacto de solución. Para ilustrar esta situación, se asume que se tienen n productos, cada uno asociado a un costo c_i y a un volumen v_i , $i \in \{1, 2, \dots, n\}$; y que se dispone de una mochila con capacidad de volumen b .

Representando con el número 1, que el producto es ingresado en la mochila y con el número 0 si no es ingresado en la mochila, al tratarse de una representación binaria, el número de combinaciones posibles está dado por 2^n .

Se trata entonces de determinar a partir de cual valor de n , el problema deja de ser una instancia pequeña del problema general. Se asume entonces que si se dispone de una máquina que pueda realizar el cómputo (evaluar y ordenar la función objetivo), de 100.000.000 (10^{+8}) de combinaciones por segundo (soluciones candidatas por segundo), el valor de n deja de ser pequeño en el caso planteado cuando $n \gg 8$; tomando $n = 60$ se tiene que el número de combinaciones dado por $2^{60} = 1,15 E^{+18}$ se computan en aproximadamente 3.7 siglos, utilizando una estrategia de búsqueda exhaustiva evaluando todas las soluciones candidatas, deja de ser razonable para lo deseado en la práctica.

En el problema de planeación de las visitas a personas privadas de la libertad con medida de aseguramiento en el domicilio, cada individuo tiene un valor de prioridad para que se le realice visita de control, denominado Peso INPEC que por analogía con el problema de la mochila es equivalente al costo del producto y el tiempo para llegar a cada persona es equivalente al volumen del producto; adicionalmente se dispone de un tiempo diario para realizar todas las visitas de control a las personas seleccionadas, esto por analogía con el problema de la mochila equivale al volumen de la mochila.

Una instancia para la planeación de las visitas a personas privadas de la libertad con medida de aseguramiento en el domicilio, en el caso de la jurisdicción de Pereira, a la fecha de este documento, es de 1000 individuos y evaluar todas las combinaciones posibles está dado por

Comentado [JCGM3]: REFERENCIAR, UNA FUENTE PUEDE SER "A Novel Artificial Bee Colony Algorithm for the Knapsack Problem" EN https://www.researchgate.net/publication/261053644_A_novel_artificial_bee_colony_algorithm_for_the_knapsack_problem

Comentado [ADBB4R3]: Resuelto

Comentado [JCGM5]: Redacción mejorada

Comentado [ADBB6R5]: ok

2^{1000} lo cual es una cantidad que supera $92E^{+281}$ veces el tiempo requerido para el caso de una instancia de 60 individuos.

Lo anterior sin tener en cuenta que el orden, en que son visitados los individuos, altera la eficiencia de la planeación de las visitas.

Por complejo que sea el problema de la planeación de visitas a personas privadas de la libertad con medida de aseguramiento en el domicilio, esta tesis tiene como objetivo ofrecer una solución de calidad aceptable hallada en un tiempo razonable.

1. TITULO

FORMULACIÓN DE UN MODELO MATEMÁTICO PARA OPTIMIZAR LA PROGRAMACIÓN DE VISITAS DE CONTROL A PRIVADOS DE LA LIBERTAD CON MEDIDA DOMICILIARIA USANDO UN ALGORITMO GENÉTICO MODIFICADO PARA EL ESTABLECIMIENTO PENITENCIARIO DE MEDIANA SEGURIDAD Y CARCELARIO DE PEREIRA

2. DEFINICIÓN DEL PROBLEMA

De acuerdo con las estadísticas oficiales del Instituto Nacional Penitenciario y Carcelario INPEC² en la última década se ha aumentado el número de personas privadas de la libertad que tienen casa por cárcel, dichas estadísticas muestran que, para el mes de junio de 2018, se supera la cifra de sesenta mil (60.000) internos se en prisión y detención domiciliaria, situación que amenaza la seguridad ciudadana. Por lo anterior se requiere mejorar los sistemas de control usados por las autoridades penitenciarias implementando mecanismos que permitan verificar el estricto cumplimiento de las penas impuestas por las autoridades judiciales.

El Instituto Nacional Penitenciario y Carcelario INPEC ha diseñado un manual interno en el cual establece y desarrolla pautas de operación, las acciones y los responsables para tramitar y dar cumplimiento a las medidas sustitutivas de Prisión y Detención Domiciliaria, Vigilancia electrónica[2]; contempla además, la creación de rutas para programar estas visitas, pero ésta programación se realiza de forma manual lo cual genera parcialidad por parte del funcionario que programa las visitas y dificulta crear rutas de acuerdo con parámetros objetivos como la gravedad del delito, la frecuencia de visitas, entre otros criterios usados por los expertos del sistema penitenciario.

Por otra parte se realizó un proceso de vigilancia tecnológica y se pudo constatar que no existen antecedentes en el sistema penitenciario colombiano del uso de algoritmos de optimización que permitan maximizar el número y peso de los internos que pueden ser visitados en una ruta[3].

² Tablero de estadístico INPEC a Mayo 15 de 2018 <http://www.inpec.gov.co/estadisticas/-/tableros-estadisticos>

3. JUSTIFICACIÓN

El problema de la planeación de visitas a personas privadas de la libertad con medida de aseguramiento en el domicilio, para la jurisdicción de Pereira se debe resolver para seleccionar de una población de 1000 individuos un grupo de ellos, maximizando la sumatoria de las prioridades para visitarlos lo cual, como se presenta en la introducción de este documento, está fuera de las capacidades de computo de los sistemas actuales si se pretende resolverlo por algún método exacto.

Se plantea realizar el modelo matemático del problema y haciendo uso de heurísticas y metaheurísticas ofrecer una solución que sea considerada de buena calidad para los expertos penitenciarios.

Atendiendo al objetivo de esta tesis, se pretende resolver el problema planteado y descrito con un modelo matemático, mediante un algoritmo genético modificado con el cual se realizará la búsqueda de los individuos a visitar en cada jornada diaria cumpliendo los requerimientos impuestos por autoridad penitenciaria, en un tiempo computacional razonable.

4. OBJETIVOS

4.1 OBJETIVO GENERAL

Formular un modelo matemático para optimizar la programación de visitas de control al cumplimiento de la pena impuesta por autoridad judicial, a privados de la libertad con medida domiciliaria usando un algoritmo genético modificado.

4.2 OBJETIVOS ESPECÍFICOS

Establecer que procedimientos y variables usan los expertos penitenciarios, para la realización de visitas de control a personas privadas de la libertad en su domicilio.

Analizar técnicas utilizadas para optimizar problemas que no se pueden resolver con métodos exactos e identificar una que pueda ser usada para ofrecer una solución de buena calidad a la programación de visitas de control a privados de libertad con detención en el domicilio.

Diseñar un modelo matemático que permita maximizar las visitas de control de las personas privadas de la libertad con domiciliaria haciendo uso de la inteligencia artificial y los algoritmos genéticos.

5. MARCO REFERENCIAL

5.1 MARCO TEÓRICO

5.1.1 Inteligencia artificial

La inteligencia artificial (IA), también llamada inteligencia computacional, es la inteligencia exhibida por máquinas. En ciencias de la computación, una máquina «inteligente» ideal es un agente racional flexible que percibe su entorno y lleva a cabo acciones que maximicen sus posibilidades de éxito en algún objetivo o tarea.[4][5][6] Coloquialmente, el término inteligencia artificial se aplica cuando una máquina imita las funciones «cognitivas» que los humanos asocian con otras mentes humanas, como por ejemplo: "aprender" y "resolver problemas". A medida que las máquinas se vuelven cada vez más capaces, tecnología que alguna vez se pensó que requería de inteligencia se elimina de la definición. Por ejemplo, el reconocimiento óptico de caracteres ya no se percibe como un ejemplo de la "inteligencia artificial" habiéndose convertido en una tecnología común. Avances tecnológicos todavía clasificados como inteligencia artificial son los sistemas capaces de jugar ajedrez, GO y manejar por sí mismos.

De acuerdo con Takeyas [7] la IA es una rama de las ciencias computacionales encargada de estudiar modelos de cómputo capaces de realizar actividades propias de los seres humanos en base a dos de sus características primordiales: el razonamiento y la conducta.

En 1956, John McCarthy [8] acuñó la expresión «inteligencia artificial», y la definió como: "...la ciencia e ingenio de hacer máquinas inteligentes, especialmente programas de cómputo inteligentes".

Para Nils John Nilsson [9] son cuatro los pilares básicos en los que se apoya la inteligencia artificial:

- Búsqueda del estado requerido en el conjunto de los estados producidos por las acciones posibles.
- Algoritmos genéticos (análogo al proceso de evolución de las cadenas de ADN).
- Redes neuronales artificiales (análogo al funcionamiento físico del cerebro de animales y humanos).
- Razonamiento mediante una lógica formal análogo al pensamiento abstracto humano.

También existen distintos tipos de percepciones y acciones, que pueden ser obtenidas y producidas, respectivamente, por sensores físicos y sensores mecánicos en máquinas, pulsos eléctricos u ópticos en computadoras, tanto como por entradas y salidas de bits de un software y su entorno software.

Varios ejemplos se encuentran en el área de control de sistemas, planificación automática, la habilidad de responder a diagnósticos y a consultas de los consumidores, reconocimiento de escritura, reconocimiento del habla y reconocimiento de patrones. Los sistemas de IA

actualmente son parte de la rutina en campos como economía, medicina, ingeniería y la milicia, y se ha usado en gran variedad de aplicaciones de software, juegos de estrategia, como ajedrez de computador, y otros videojuegos.

5.2 MARCO CONCEPTUAL

5.2.1 Algoritmos genéticos

Los Algoritmos Genéticos (AG's) son métodos de optimización basados en una simulación parcial de los mecanismos de la evolución natural. Están basados en la teoría de la evolución que surge con las investigaciones de Charles Darwin [10]. Los algoritmos genéticos fueron creados en la década de los 60's por John Holland [11] como un modelo para el estudio del fenómeno de adaptación natural y para el desarrollo de mecanismos que permitieran incorporar este fenómeno a los sistemas de cómputo. Los algoritmos genéticos alcanzaron popularidad a raíz de la publicación del libro de Goldberg [12] que ponía las bases fuertes para su aplicación en problemas prácticos. Los algoritmos evolutivos constituyen una parte importante de la Computación Evolutiva, un área de la Inteligencia Artificial en constante crecimiento. Actualmente son innumerables las aplicaciones exitosas en las más diversas áreas industriales, comerciales y de la ingeniería [12][13][14].

Comentado [JCGM7]: Referenciar este libro ¿Cuál es?

Comentado [ADBB8R7]: ok

5.2.2 Heurística

Una heurística es una técnica que, al aplicarse a un problema, en el caso de este documento a un problema de optimización, reduce la complejidad computacional del problema sacrificando completitud, ya que se enfoca en explorar una parte menor del espacio de soluciones del problema.

Existen una gran variedad de técnicas heurísticas sin que exista una única técnica que sea universal; estas dependen del contexto del problema [15], del conocimiento de expertos, del sentido común, de los recursos disponibles, de la calidad requerida en la solución del problema, entre otras.

Las técnicas heurísticas están basadas en elementos, uno de ellos, por ejemplo, en la estrategia Divide y vencerás [16], en las que el problema general se divide en instancias más pequeñas del problema, resolviendo cada una de estas se puede obtener una solución aceptable del problema general. Otro elemento que se aprovecha es la de añadir componentes a la solución hallada, paso a paso, buscando el máximo beneficio en cada paso, un ejemplo de aplicación de este elemento se puede encontrar en los algoritmos voraces (*greedy*) [16].

La reducción del problema, como elemento de la heurística, permite al simplificarlo, identificar alguna característica que sea parte de lo que debe poseer la solución óptima.

Otro elemento de las técnicas heurísticas es simplificar el problema mediante la manipulación del modelo, con lo cual en un esquema teórico reducido se pueden hallar las variables de decisión con menor esfuerzo y a partir de estas hallar la solución general del

Comentado [JCGM11]: REFERENCIAR: Judgment under Uncertainty: Heuristics and Biases Amos Tversky; Daniel Kahneman Science, New Series, Vol. 185, No. 4157. (Sep. 27, 1974), pp. 1124-1131 <http://psiexp.ss.uci.edu/research/teaching/Tversky_Kahneman_1974.pdf>

Comentado [ADBB12R11]: ok

Comentado [JCGM13]: INSERTAR REFERENCIA: INTRODUCTION TO ALGORITHMS THIRD EDITION THOMAS H. CORMEN, CHARLES E. LEISERSON, RONALD L. RIVEST, CLIFFORD STEIN 2.3.1 The divide-and-conquer approach PARA los datos del libro, este se puede descargar en PDF desde <<http://labs.xjtudlc.com/labs/wldmt/reading%20list/books/Algorithms%20and%20optimization/Introduction%20to%20Algorithms.pdf>>

Comentado [ADBB14R13]: ok

Comentado [JCGM15]: INSERTAR REFERENCIA AL CAPITULO 16 TAMBIEN DE: INTRODUCTION TO ALGORITHMS THIRD EDITION THOMAS H. CORMEN, CHARLES E. LEISERSON, RONALD L. RIVEST, CLIFFORD STEIN PARA los datos del libro, este se puede descargar en PDF desde <<http://labs.xjtudlc.com/labs/wldmt/reading%20list/books/Algorithms%20and%20optimization/Introduction%20to%20Algorithms.pdf>>

Comentado [ADBB16R15]: ok

problema. Utilizar criterios de terminación o parada hacen también parte de las técnicas heurísticas.

Aplicando una o varias heurísticas que estén relacionadas con la función a optimizar, permiten hallar alguna o varias soluciones del problema, sin que esto signifique que esté o no relacionadas con el objetivo de solución del problema. Por lo tanto, la estructura propia del modelo matemático y como se combinen las técnicas son los que determinan la calidad de las soluciones.

Para el problema de planeación de visitas a personas privadas de la libertad con medida de aseguramiento en el domicilio para una jurisdicción responsable de 1000 individuos, separados entre ellos por distancias esparcidas en un departamento como Risaralda, que son visitados en un lapso de tiempo diario de 8 horas, que tienen una prioridad de visita impuesta por autoridad penitenciaria, que pueden ser visitados mas de una vez en una misma jornada y que rigurosamente deben ser seleccionados en forma aleatoria, de forma tal que la suma total de los pesos de prioridad de visita sean máximos; como se muestra en la introducción, conlleva, si se desea una solución exacta, un esfuerzo computacional fuera de lo razonable.

Es justo un problema combinatorio que puede resolverse mediante la aplicación de técnicas heurísticas, en la que *se reduce el espacio de soluciones*, cumpliendo con la aleatoriedad, de estricto cumplimiento, requerida por la autoridad penitenciaria, seleccionando un individuo al azar y hallando las distancias a un grupo de individuos cercanos, mediante una heurística también de reducción que corresponde a un porcentaje de la población total.

Determinar el tamaño del grupo, se logra mediante una heurística de manipulación del modelo en la que se establece un periodo Q , en días, en el cual se estima estén visitados todos los individuos.

5.2.3 Metaheurística

De acuerdo con J. P. y I. H. Kelly [17], las técnicas metaheurísticas se definen como "... una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que los métodos heurísticos clásicos no son efectivos ni eficientes. Los metaheurísticos proporcionan un marco general para crear nuevos algoritmos híbridos combinando nuevos conceptos derivados de la inteligencia artificial, la evolución biológica y la mecánica estadística".

Estas técnicas tienen como objetivo enfocarse en tratar de hallar soluciones confiables y en tiempo razonable, sin pretender hallar la solución óptima; en ocasiones estas técnicas pueden sacar provecho de soluciones consideradas de mala calidad que pueden servir para salir de soluciones óptimas locales.

Algunas de las metaheurísticas realizan la búsqueda de soluciones en estrategias basadas en la naturaleza.

La aplicación de las técnicas metaheurísticas requiere que se guarde especial atención y cuidado en la estrategia para codificar las diversas alternativas que surgen en el espacio de búsqueda definido para el problema.

Es necesario también que, en caso de usarse, la función de adaptación esté bien definida, es decir de forma tal que represente a la función objetivo original. Adicionalmente, la solución con la que se inicie el proceso, en el caso de problemas de múltiples nodos (multimodales), debe estar en amplio sentido correlacionada con la calidad de la solución que se halle al final del proceso.

Se debe contar con un criterio de parada que determina en que momento se da por culminado el proceso de búsqueda de soluciones; es común que el criterio de parada se dé cuando una solución vecina no es mejor que la solución actual; también es válido un criterio de parada en el cual el proceso se detenga cuando luego de transcurrido un número de iteraciones, no halla una mejora en la solución encontrada.

Para el caso del problema objeto de esta tesis, se toma como base la metaheurística de Algoritmos Genéticos, estos algoritmos hacen evolucionar una población de individuos sometiéndola a acciones aleatorias parecidas a las que actúan en la evolución biológica (mutaciones y recombinaciones genéticas), así como también a una selección de acuerdo con algún criterio, en función del cual se decide cuáles son los individuos más adaptados, que sobreviven, y cuáles los menos aptos, que son descartados[12].

6. DESAROLLO METODOLÓGICO

De acuerdo con la información que se recogió para responder al problema de investigación el estudio será de tipo experimental de enfoque cuantitativo. Para llevar a cabo el objeto de estudio el desarrollo metodológico se dividió en 3 fases, las cuales se describen a continuación:

1. En la fase inicial del proceso se hizo necesario establecer qué procedimientos y variables usan los expertos penitenciarios, para la realización de visitas de control a reclusos con detención en domicilio, dicha información se obtuvo aplicando instrumentos de entrevista al personal de la guardia penitenciaria encargada de realizar estos procesos en los establecimientos penitenciarios del departamento de Risaralda y revisando los manuales, procedimientos, directivas y demás documentos de doctrina institucional propios del sistema penitenciario donde está documentada esta labor. El detalle de esta fase puede visualizarse en el capítulo I del presente documento.
2. En la segunda fase se hizo necesario estudiar las técnicas utilizadas para optimizar problemas que no se pueden resolver con métodos exactos y se identificó que las técnicas heurísticas y metaheurísticas son las que mejor se adaptan de acuerdo con las variables del problema de estudio obtenidas en la fase anterior. El detalle de esta fase puede visualizarse en el marco de referencia del presente documento.
3. En la tercer y última fase se diseñó un modelo matemático que permite maximizar las visitas de control de las personas privadas de la libertad con domiciliaria haciendo uso de la inteligencia artificial y los algoritmos genéticos. La función objetivo permitió maximizar el número de privados de libertad que pueden ser visitados en un periodo de tiempo, dando prelación a visitar a aquellos internos que representan mayor peso, ya sea por los delitos cometidos, perfil delincucional o requerimiento legal. El detalle de esta fase puede visualizarse en el capítulo II del presente documento.

De la misma manera se diseñó un algoritmo genético, el cual fue implementado en el lenguaje de programación python para realizar las pruebas y afinamiento del algoritmo, se usaron 100 registros generados de forma aleatoria donde sus coordenadas geográficas están ubicadas en la ciudad de Pereira y el peso o prioridad asignada a nodos, oscila en valores de 1 a 100 los cuales indican la prioridad a ser visitados, siendo 1 el menos prioritario y 100 el más prioritario.

7. CAPÍTULO I

Este capítulo tiene como finalidad conocer los procedimientos y variables que usan los expertos penitenciarios para la realización de visitas de control a personas privadas de la libertad en su domicilio. Para determinar lo anterior se realizó una verificación del marco legal que regula las penas y medidas de seguridad de las personas beneficiadas de prisión o detención domiciliaria.

De acuerdo con el Ministerio de Justicia y del Derecho, las normas que regulan las medidas de Reclusión Domiciliaria, Prisión Domiciliaria y Vigilancia Electrónica de acuerdo con el documento de “subrogados penales, mecanismos sustitutivos de pena y vigilancia electrónica en el sistema penal colombiano” son las siguientes:

Reclusión domiciliaria: Se encuentra regulada en el artículo 68 del Código Penal, Ley 599 de 2000.

Prisión domiciliaria: La prisión domiciliaria como mecanismo sustitutivo de la prisión se encuentra regulada en el artículo 38 del Código Penal, Ley 599 de 2000.

Vigilancia electrónica: El Sistema de Vigilancia Electrónica en Colombia surge en el año 2004. En ejercicio de las facultades extraordinarias conferidas por el Acto Legislativo 03 de 2002, el presidente de la República expidió el Decreto 2636 de 2004 mediante el cual se hace alusión por primera vez a los Sistemas de Vigilancia Electrónica en Colombia. Mediante este Decreto se establece como competencia de los jueces de ejecución de penas la imposición de medidas de Vigilancia Electrónica como mecanismo sustitutivo de la prisión en casos de delitos considerados menores, es decir, con pena impuesta no superior a cuatro años de prisión.

Por su parte, la Ley 906 de 2004 estableció la posibilidad de imponer una modalidad de Vigilancia Electrónica como medida de aseguramiento no privativa de la libertad o como mecanismos de garantía al cumplimiento de la detención domiciliaria de procesados si se considera que la residencia es suficiente para cumplir con los fines de la medida de aseguramiento, si el imputado es mayor de 65 años, padece grave enfermedad, es madre o padre cabeza de familia o si la detenida se encuentra en los últimos meses del embarazo.

Por otro lado, el Instituto Nacional Penitenciario y Carcelario INPEC ha creado en el 2010 el manual de prisión y detención domiciliaria el cual establece los parámetros a seguir para operacionalizar a nivel Nacional el manejo, control y vigilancia de la medida sustitutiva de prisión y detención domiciliaria en los establecimientos de reclusión de Orden Nacional. Inicia con el recibimiento de la orden judicial y culmina con la libertad del interno[18].

En este Manual se reglamenta en el apartado número 16, el control del cumplimiento de la medida y programación de visitas que describe que el servidor público encargado de la sección de domiciliaras, adoptará como medidas para el control y cumplimiento de la prisión o detención domiciliaria la programación diaria de visitas aleatorias, así como también podrá adoptar las siguientes:

1. Uso de medios de comunicación como llamadas telefónicas.
2. Testimonios de vecinos y allegados.
3. Labores de inteligencia.

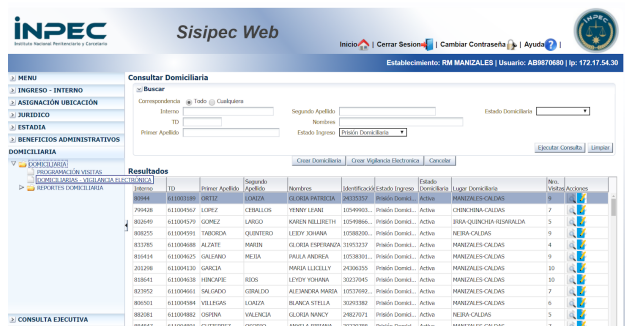
Para la programación de visitas aleatorias el servidor público encargado de la sección de domiciliarias hará uso del aplicativo del SISIPPEC web, modulo estadía-domiciliaria - programación visitas domiciliarias para tal fin. La programación de visitas aleatorias se hará con el aplicativo SISIPPEC web, teniendo en cuenta el plan de ordenamiento territorial de cada ciudad o municipio (localidades, barrios, corregimientos o veredas) Así:

Ingresa al aplicativo Sisippec Web modulo - Estadía-Domiciliaria-Programación visitas domiciliarias, diligencia fecha, lugar, ingreso (se digita N.I del interno), fecha salida, nombres y apellidos del responsable (Unidad de Guardia responsable e efectuar la visita) y en la casilla de consideraciones se registra el motivo para el cual se programa la visita se guarda la información y una vez se genera el número del programa, se imprime la planilla a través del modulo Estadía-domiciliaria-reporte programa domiciliaria[2].

7.1 EXPERTOS PENITENCIARIOS:

Con el fin de conocer el proceso operativo que se realiza en virtud de las visitas domiciliarias, se entrevistaron diferentes funcionarios del cuerpo de custodia y vigilancia del INPEC, quienes se encargan de planear, ejecutar y reportar el resultado de las programaciones de visita a los privados de libertad.

Ilustración 1. Aplicativo Sisippec Web INPEC



Fuente: Autores

En la ilustración 1, se observa el sistema de información Sisippec Web, en el cual se gestiona toda la información asociada a las personas privadas de la libertad a cargo del INPEC, en este aplicativo se hace la programación de los planes de visita. Sin embargo, la escogencia de los candidatos a ser visitados se define de forma manual. El oficial Jorge Osorio, funcionario del INPEC miembro del Cuerpo de Custodia y Vigilancia en el grado de Oficial Logístico asegura que esta es una tarea lenta, poco óptima y que se presta para sesgos en el proceso.

De otra parte se realizó entrevista (Anexo 2) al funcionario penitenciario Michael Mahecha Rúa, quien labora en la Dirección Regional del INPEC teniendo a cargo el proceso de control de personas privadas de la libertad con domiciliaria, este funcionario ratifica lo mencionado por el oficial logístico Jorge Osorio, y hace énfasis en la importancia de las visitas domiciliarias para el control del cumplimiento a la medida impuesta por la autoridad judicial; además, reitera la importancia de realizar dichas visitas de forma periódica, aleatoria y se debe usar el aplicativo Sisippec Web para esta tarea.

ción 2, se observa imagen de un prototipo de software propuesto por el INPEC Nacional de Colombia para mejorar el sistema de visitas, para este caso se desea un sistema de información geográfica, la ubicación del domicilio asignado a la persona con detención extramural a cargo del INPEC, con el fin de facilitar el control y la toma de decisiones sobre este personal.

29

8. CAPÍTULO II

En este capítulo se muestra el diseño de un modelo matemático que permite maximizar las visitas de control de las personas privadas de la libertad con domiciliaria, haciendo uso de un algoritmo genético.

8.1 NOMENCLATURA

En esta sección, se incluyen la notación utilizada en el desarrollo del documento, así como también los símbolos generales y abreviaturas utilizadas en las diferentes partes de la presente Tesis de Maestría.

Tabla 1 Variables del Modelo Matemático

| No. | Nombre | Descripción | Definición | Tipo | Ejemplo |
|-------|--------------------------|--|---|-----------|---|
| (8.1) | PPL | Conjunto de personas privadas de la libertad en el domicilio | Nombre dado por el INPEC a la población en prisión o detención domiciliaria con o sin vigilancia electrónica | Abrev. | Jurisdicción Pereira |
| (8.2) | PL_i | Persona privada de la libertad en el domicilio | Nombre dado por el INPEC a una persona en prisión o detención domiciliaria con o sin vigilancia electrónica | Abrev. | $PL_i \in PPL$ |
| (8.3) | n | Número total de personas privadas de la libertad (PPL) | Se refiere a la cantidad de personas asignadas a una jurisdicción del INPEC | Parámetro | 1000, 2500, ... |
| (8.4) | P | Peso INPEC | Valor entre 1 y 100 asignado por el INPEC a cada persona privada de la libertad para indicar la prioridad de visita | Parámetro | 6, 21, ..., 92, ... |
| (8.5) | v | Velocidad de desplazamiento | Valor que se determina como seguro y adecuado para transportarse entre las personas a visitar | Parámetro | $40 \frac{km}{h}, 60 \frac{km}{h}, ...$ |

| No. | Nombre | Descripción | Definición | Tipo | Ejemplo |
|--------|----------------|---|--|-----------|--|
| (8.6) | d_{ij} | Distancia entre el nodo anterior i al nodo siguiente j | Distancia calculada a partir de las coordenadas de georreferenciación entre ³ $PL_i(\delta_i, \lambda_i)$ y $PL_j(\delta_j, \lambda_j)$ δ : Latitud; λ : Longitud | Cálculo | |
| (8.7) | t_{ij} | Tiempo para ir desde el nodo anterior i al nodo siguiente j | Se estima a partir de la distancia entre dos nodos contiguos a una velocidad constante | Cálculo | $t_{ij} = \frac{d_{ij}}{v}$ |
| (8.8) | dt_j | Tiempo asignado para la toma de información de PL_j | Valor determinado por el INPEC como suficiente para tomar los datos relacionadas con el objeto de la visita a una persona privada de la libertad | Parámetro | 10 minutos, 15 minutos, ... |
| (8.9) | T_j | Tiempo para atender a PL_j | Es la suma del tiempo t_{ij} más el tiempo asignado para la toma de información dt_j | Cálculo | $T_j = t_{ij} + dt_j$ |
| (8.10) | k | Número de una única persona seleccionada al azar por jornada diaria | Persona que se selecciona al azar y a la que se debe obligatoriamente realizar visita en la jornada | Cálculo | $k \in PPL$ |
| (8.11) | <i>periodo</i> | Tiempo total requerido para visitar toda la población PPL | Tiempo en cual se estima que todo PL este visitado | Parámetro | 10 días, 45 días, ... |
| (8.12) | Q | Cantidad estimada de personas a visitar por | Se aproxima a entero con truncamiento por el techo | Cálculo | $\left\lceil \frac{n}{periodo} \right\rceil$ |

Comentado [JCGM17]: CORREGIDO, Y ESTA OK

³ Cálculo de la distancia entre dos puntos a partir de las coordenadas de latitud y longitud: [Weisstein, Eric W. "Great Circle." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/GreatCircle.html](http://mathworld.wolfram.com/GreatCircle.html)

| No. | Nombre | Descripción | Definición | Tipo | Ejemplo |
|--------|-----------|---|--|-----------|----------------------------------|
| | | jornada diaria | | | |
| (8.13) | f_{min} | Factor para determinar el mínimo de personas a visitar por jornada aceptables | Es un valor entre 0 y 1 | parámetro | $0 < f_{min} \leq 1$ |
| (8.14) | Q_{min} | Valor mínimo aceptado, de personas a visitar | $0 < Q_{min} \leq Q$ | Cálculo | $Q_{min} = Q * f_{min}$ |
| (8.15) | T_{o1} | Tiempo requerido para llegar desde el nodo origen al primero nodo a visitar | Siempre, en cada jornada, se inicia la visita a las Q personas seleccionadas, desde el nodo origen | Cálculo | $T_{o1} = \frac{d_{o1}}{v}$ |
| (8.16) | T_{eo} | Tiempo requerido para llegar desde el último nodo visitado al origen | Siempre, en cada jornada, se culmina la visita a las Q personas seleccionadas, en el nodo origen | Cálculo | $T_{eo} = \frac{d_{eo}}{v}$ |
| (8.17) | b | Tiempo total disponible para visitar los nodos de la ruta | Es el tiempo de la jornada laboral asignado al funcionario encargado de realizar las visitas | Parámetro | 480 minutos, 650 minutos, ... |
| (8.18) | S | Cantidad de permutaciones de tamaño Q que se obtienen de la población total PPL | Q -permutaciones de n . Secuencia ordena de Q individuos, con ningún individuo apareciendo más de una vez en la secuencia, del conjunto de los n individuos de PPL | Cálculo | $S = \frac{n!}{(n-Q)!}$ [16] |

Fuente: Autores

8.2 ANÁLISIS DEL ESFUERZO COMPUTACIONAL

Programar las visitas de control al cumplimiento de la pena impuesta por autoridad judicial a privados de la libertad con medida domiciliaria implica: seleccionar desde un conjunto de n individuos (8.3), un subconjunto **aleatorio** del conjunto *PPL* (8.1), de tamaño Q individuos (8.12), donde Q individuos es el cociente entre el total de individuos (8.3) y el período (8.11) de tiempo en el que se desea que todos los individuos hayan sido visitados. Escogidos de forma tal que la selección de tamaño Q tenga el peso máximo sujeto a que puedan ser visitados en el tiempo b (8.17) disponible para realizar el recorrido de las visitas incluyendo los tiempos desde el origen a la primera persona a visitar (8.15) y el tiempo desde la última persona visitada hasta en lugar de retorno (8.16), más el tiempo (8.9) de toma de datos para verificación de datos de las personas visitadas que están cumpliendo con la medida de aseguramiento en el momento de la visita.

La representación matemática del esfuerzo computacional se presenta a continuación:

n : Total de individuos (8.3) de la población

periodo: Tiempo total (8.11) requerido para que el total de individuos de la población (8.3), hayan sido visitados

Q : Cantidad de personas a visitar (8.12) por jornada diaria

$$Q = \left\lfloor \frac{n}{\text{periodo}} \right\rfloor$$

S : Cantidad de subconjuntos ordenados (8.18), de Q individuos del conjunto de los n individuos de *PPL*

$$S = \frac{n!}{(n - Q)!}$$

Encontrar una solución óptima podría lograrse hallando todas las soluciones candidatas, en este caso S , calcular la función objetivo para cada una de ellas y calcular el tiempo necesario para recorrerlas.

Tomando como ejemplo el caso de la ciudad de Pereira, asumiendo que $n = 1000$, (Al año 2018 esta cifra está superada), y un *periodo de* 30 días para que todos los individuos estén visitados, se tiene que:

$$\begin{aligned} Q &= \left\lfloor \frac{1000}{30} \right\rfloor = 34 \\ S &= \frac{1000!}{(1000 - 34)!} \\ S &= \frac{1000!}{966!} = 5,67 E^{+101} \end{aligned}$$

Suponiendo que existe una computadora que puede ordenar y evaluar la función objetivo y calcular la ruta de menor tiempo, de cien millones (10^8) de soluciones por segundo, el tiempo para hallar la solución óptima es de 1.8×10^{92} siglos aproximadamente, en el caso extremo con un *periodo* = 1 (días), se tiene $Q = 1000$ y $S = 1000!$ Esto es 4×10^{2567} soluciones candidatas.

El esfuerzo computacional crece de manera exponencial con el tamaño del problema lo que es imposible de resolver con la tecnología disponible a la fecha de este documento.

Se infiere de (8.18) que se está frente a un problema combinatorio que requiere solución en tiempo razonable que permita hallar una solución valida, denominada óptimo local o relativo.

Teniendo presente que se debe dar solución al problema, se plantea un modelo matemático a partir del cual y mediante técnicas heurísticas y metaheurísticas, se pueda hallar un óptimo local.

8.3 MODELO MATEMÁTICO

Existe una población de n personas (8.3), privadas de la libertad (8.1), cada persona (8.2) tiene un Peso de prioridad para visitarlo asignado por el INPEC (8.4), y un tiempo (8.9) para llegar a cada uno y visitarlo tomando los datos de verificación requeridos por el INPEC, esto es un tiempo necesario (8.7) para llegar desde un nodo anterior al siguiente más un tiempo para toma de datos (8.8) durante la visita.

Se desea visitar algunas de las personas (8.12) en forma **aleatoria**, en un período de tiempo b (8.17) de tal forma que la sumatoria del peso (8.4), de cada persona a visitar sea máximo.

Sean $X_j = \{0, 1\}$ las variables de decisión en donde $X_j = 1$, significa que la persona (8.2) es seleccionada para visitarse.

Función objetivo:

$$\text{Max } z(X) = \left(\sum_{j=1}^{n-1} P_j X_j \right) + P_n X_n ; P_j \in \{1, 2, \dots, 100\}; P_k \in P_j \quad (8.19)$$

Sujeto a:

$$T_{o1} + \left(\sum_{j=1}^n T_j X_j \right) + T_{eo} \leq b; \quad (8.20)$$

T_{o1} : *Tiempo desde el origen al primer nodo a visitar*

T_{eo} : *Tiempo desde el último nodo visitado al origen*

b: Tiempo total disponible para visitar los nodos de la ruta

$$X_j \in \{0, 1\}; j = 1, 2, \dots, k, \dots \quad (8.21)$$

$$k \text{ aleatorio}; k \in \{1, 2, \dots, n\} \quad (8.22)$$

$$Q * f_{min} \leq \sum_{j=1}^n X_j \leq Q; 0 < f_{min} \leq 1 \quad (8.23)$$

$$Q = \left\lceil \frac{n}{periodo} \right\rceil \quad (8.24)$$

8.4 HEURÍSTICAS PARA LA EJECUCIÓN DEL ALGORITMO

Sea F el número de personas asignadas por el INPEC para realizar las vistas en la jornada, entonces:

Para $F = 1$

- Introducir los parámetros: $v, T_j, periodo, f_{min}$ (8.5), (8.9), (8.11), (8.13)
- Calcular los valores relacionados con los parámetros introducidos
- Seleccionar un nodo k aleatoriamente. Se puede utilizar el método de balotas o contratar un servicio como el de *RANDOM.org*
- Aplicar un algoritmo genético modificado para $F = 1$ con toda la población (8.1)

Si F aumenta:

Aplicar el algoritmo genético modificado tantas veces como funcionarios sean asignados por el INPEC en cada jornada, teniendo siempre en cuenta incluir a toda la población; la ley permite que una misma persona (8.2) sea visitada más de una vez en una misma jornada.

8.5 CODIFICACIÓN DEL PROBLEMA

Para representar en forma consistente una propuesta de solución del problema, objeto de este documento, en su espacio de soluciones, se ha escogido la codificación binaria desde la cual es posible encontrar el valor de la función objetiva del problema y calcular el orden en que los individuos (nodos) son visitados es decir la ruta para las visitas.

Cada individuo (8.2) de la población (8.1) se representa por un nodo numerado desde 1 hasta el total de la población de la jurisdicción determinada por el INPEC (8.3).

Una Q-permutación (8.18) de individuos, de tamaño Q , que pertenece a S (8.18), puede incluir a todos o únicamente algunos individuos para pertenecer a la lista de individuos a visitar.

Si el individuo que pertenece a Q (8.12) no está incluido en la ruta, se representa por el número “0” y sin el individuo que pertenece a Q , si está incluido en la ruta, se representa por el número del nodo.

$$X \in \{0, PL_i\} \quad (8.25)$$

El tiempo entre dos nodos i e j t_{ij} se calcula a partir de la distancia entre cada par de nodos consecutivos (8.6) a partir de una velocidad promedio de desplazamiento v (8.5) suministrado como parámetro de entrada al algoritmo genético, mediante la fórmula:

$$t_{ij} = \frac{d_{ij}}{v} \quad (8.26)$$

Dado que la norma penitenciaria indica que las visitas se deben realizar en forma aleatoria, se selecciona un nodo aleatorio denominado k (8.10) que pertenece al conjunto de personas privadas de la libertad en la jurisdicción determinada por el INPEC (8.1) y que siempre debe pertenecer a Q_{min} (8.14)

La codificación queda formada por {nodo origen, Q candidato, nodo llegada}

Q =

| | | | | | | | | | |
|--------|---|-----|---|-----|----------------|-----|---|---|---------|
| origen | a | ... | m | ... | K aleatorio | ... | 0 | z | llegada |
|--------|---|-----|---|-----|----------------|-----|---|---|---------|

Los valores $a, \dots, m, \dots, K, \dots, z$; representan los nodos que sí son seleccionados para la ruta de visita, 0 representa los nodos que no son seleccionados para la ruta de visita.

Ejemplo.

Para un conjunto Q de longitud 8, que pertenece a S y numerada con enteros positivos desde 1 hasta 1000, y cuyos elementos son en orden {943, 36, 5, 0, 0, 3, 824} y el K aleatorio es 3, se tiene:

Q =

| | | | | | | | | | |
|--------|-----|----|---|---|---|-----|---|-----|---------|
| origen | 943 | 36 | 5 | 0 | 0 | ... | 3 | 824 | llegada |
|--------|-----|----|---|---|---|-----|---|-----|---------|

9. ALGORÍTMO

9.1 PSEUDOCÓDIGO

A continuación se presenta en pseudocódigo, la descripción de alto nivel compacta e informal del principio operativo del algoritmo genético modificado.

Dentro de las modificaciones a destacar se debe tener en cuenta que en este algoritmo la elección de un individuo al azar como criterio de inicio es fundamental, debido a que los nodos seleccionados para generar la población inicial serán los más cercanos al individuo seleccionado, de acuerdo a criterio de ejecución

```
1      Inicia /*Algoritmo Genético Modificado*/
2          Elegir individuo  $k$  al Azar
3          Buscar más cercanos al individuo  $k$ 
4          Generar una población inicial.
5          Computar la función objetivo de cada individuo.
6          Mientras No Terminado Hacer
7              Inicia /* Producir nueva generación */
8                  Para Tamaño población/2 Hacer
9                      Inicia /*Ciclo Reproductivo */
10                         Seleccionar dos individuos de la anterior generación, para el
                           cruce (probabilidad de selección proporcional a la función de
                           evaluación del individuo).
11                         Cruzar con cierta probabilidad los dos individuos obteniendo
                           dos descendientes.
12                         Mutar los dos descendientes con cierta probabilidad.
13                         Computar la función de evaluación de los dos descendientes
                           mutados.
14                         Reemplazar los dos descendientes mutados en la nueva
                           generación.
15                     Termina
16                     Si la población ha convergido Entonces
17                         Terminado:= Verdadero
18                     Termina
19      Termina
```

9.2 PYTHON

El algoritmo de este proyecto se ha implementado en el lenguaje Python, este es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo

nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Es un lenguaje similar a Perl, pero con una sintaxis muy limpia y que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos [19].

9.3 OPERACIÓN DEL ALGORITMO GENÉTICO

Tras parametrizar el problema en una serie de variables, se codifican en un cromosoma. Todos los operadores utilizados por un algoritmo genético se aplicarán sobre estos cromosomas o sobre poblaciones de ellos. En el algoritmo genético va implícito el método para resolver el problema.

Las soluciones codificadas en el cromosoma, compiten con el método de torneo para ver cuál constituye la mejor solución (aunque no necesariamente la mejor de todas las soluciones posibles) ejerciendo presión selectiva sobre la población, de forma que sólo los mejor adaptados (aquellos que resuelvan mejor el problema) sobrevivan o leguen su material genético a las siguientes generaciones.

9.3.1 Tamaño de la Población

De otra parte en la implementación de este algoritmo se genera una población con un límite de 2000 individuos, donde se permite un máximo del 10% de soluciones no factibles.

9.3.2 Codificación de las Variables

Los cromosomas contienen información acerca de la solución que representa. La codificación se puede realizar de varias formas; la más utilizada es mediante una cadena de números binarios (1s o 0s), pero también se puede realizar la codificación mediante números enteros o incluso cadenas de palabras.

La elección de la codificación depende del problema a resolver, pues puede darse la situación en la que la resolución de un caso sea más óptimo el uso de una codificación basada en números reales.

9.3.3 Codificación Numérica

En este algoritmo se utilizan cadenas de números que representan un número en una secuencia; este tipo de codificación resulta muy útil cuando se utiliza en problemas en los que hay que ordenar algo. Un factor determinante es que no solo se requiere saber si el individuo va incluido en la ruta o no, sino que se requiere saber el orden, ya que esto puede afectar el cumplimiento de las restricciones asociadas al tiempo.

Esta cadena representa un individuo, donde los ceros al inicio y fin indican el punto de partida y de llegada, el alelo marcado en negrita muestra el elemento que va fijo en la ruta, aunque no en qué orden.

| | | | | | | | | | | | |
|---|----|----|----|----|----|----|----|----|----|----|---|
| 0 | 69 | 84 | 86 | 25 | 26 | 51 | 86 | 27 | 35 | 76 | 0 |
|---|----|----|----|----|----|----|----|----|----|----|---|

Esta cadena se indica; 0: indica que se inicia y se termina en el nodo cero, los genes con alelos vacíos indican que no se asignan a la ruta y los genes que tienen valores numéricos indican que el elemento va en la ruta y cuál es el orden.

| | | | | | | | | | | | |
|---|---|---|--|--|---|---|--|--|---|--|---|
| 0 | 5 | 3 | | | 4 | 2 | | | 1 | | 0 |
|---|---|---|--|--|---|---|--|--|---|--|---|

9.3.4 Selección por Rueda de Ruleta

Se hace necesario hacer una selección con los individuos más capacitados para que éstos sean los que se reproduzcan con más probabilidad de acuerdo con la teoría de Darwin, en la cual, los más capacitados son los que deben sobrevivir y crear una nueva descendencia más facultada.

Por lo tanto una vez evaluado cada cromosoma y obtenida su aptitud, se tiene que crear la nueva población teniendo en cuenta que los buenos rasgos de los mejores se transmitan a ésta.

Esta selección se puede realizar de varias formas, sin embargo, en esta implementación se hace uso de la selección por torneo. Se escogen de forma aleatoria cuatro (4) individuos de la población y el que tiene mayor puntuación se reproduce, sustituyendo su descendencia al que tiene menor puntuación.

9.3.5 Reproducción o Crossover

Una vez se realiza la selección de los cromosomas se procede a realizar la reproducción o cruce entre dos de estos cromosomas, el cual consiste en el intercambio de material genético entre dos cromosomas. El objetivo del cruce es conseguir que el descendiente mejore la aptitud de sus padres.

Se fija una tasa de cruzamientos la cual indica la frecuencia con la que se producen cruces entre los cromosomas padre, es decir, que haya probabilidad de reproducción entre ellos. En caso de que no exista probabilidad de reproducción, los hijos serán copias exactas de los padres. En caso de haberla, los hijos tendrán partes de los cromosomas de los padres[20].

Para garantizar una tasa de recombinación del 30%, lo cual indica que los hijos tienen una probabilidad de recibir dicho porcentaje de genes de sus padres, usando el método *Partially Mapped Crossover* (PMX) [21] el cual funciona de la siguiente manera:

Procedimiento informal para padres P1 y P2:

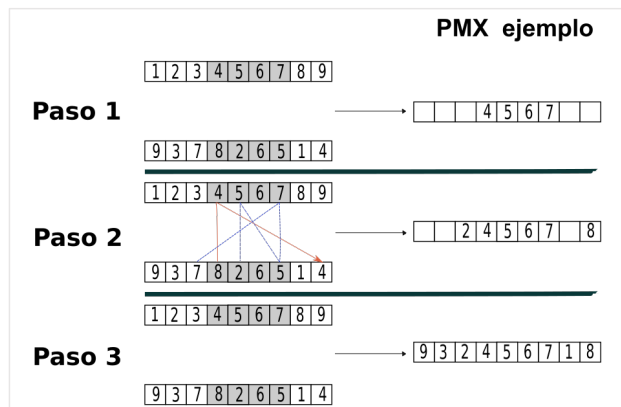
1. Copiar un segmento al azar de P1.

Comentado [JCGM18]: INSERTAR REFERENCIA: PODRÍA SER DESDE
<<https://www.hindawi.com/journals/cin/2017/7430125/>>

Comentado [ADBB19R18]: Ok

2. Comenzando en el primer punto de crossover, buscar elementos en ese segmento de P2 que no han sido copiados.
 3. Para cada uno de esos i buscar en el hijo, que elementos j han sido copiados en su lugar.
 4. Poner i en la posición ocupada por j en P2, porque sabemos que no pondremos j allí ya que j ya fue copiado.
 5. Si el lugar ocupado por j en P2 ya ha sido ocupado en el hijo en k, poner i en la posición ocupada por k en P2.
 6. El resto puede ser copiado de P2.
- Se hace lo mismo para el otro hijo.

Ilustración 3. Ejemplo Método (PMX) Partially Mapped Crossover



Fuente: Curso de Postgrado – UC3M – Junio 16,17,18 - 2004⁴

9.3.6 Mutación

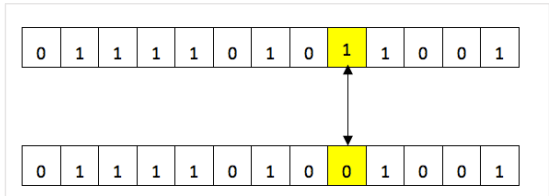
Tras el cruce, tiene lugar la mutación, en términos de evolución las mutaciones suelen en promedio ser beneficiosas, pues contribuyen a la diversidad genética de la especie. Además, previenen a las soluciones de la población de verse limitadas por un óptimo local. Por lo tanto, la mutación consiste en modificar ciertos genes de forma aleatoria, atendiendo a la probabilidad de mutación establecida con anterioridad. La mutación depende de la codificación y de la reproducción.

⁴ Imagen basada en el Curso de Postgrado – UC3M de Algoritmos Evolutivos y Meméticos <http://www.exa.unicen.edu.ar/escuelapav/cursos/bio/l3.pdf> Consultada el 01/05/2018

La tasa de mutación indica la frecuencia con la que los genes de un cromosoma son mutados. Si no hay mutación, los descendientes son los mismos que había tras la reproducción. En caso de que haya mutaciones, parte del cromosoma descendiente es modificado y si la probabilidad de mutación es del 100%, la totalidad del cromosoma se cambia. En este caso, se selecciona una cantidad determinada de individuos según la tasa de mutación y se les intercambia dos posiciones de manera aleatoria.

La tasa de mutación usada en la prueba del algoritmo actual es igual a un 20%, pero puede ser ajustada en los parámetros iniciales de ejecución del algoritmo.

Ilustración 4. Mutación



Fuente: Autores

9.3.7 Reemplazo

Una vez aplicados los operadores genéticos, se seleccionan los mejores individuos para conformar la población de la generación siguiente.

9.3.8 Criterio de parada

Los criterios de parada dispuestos para el algoritmo son dos, el primero de ellos por tiempo de ejecución y el segundo por número de iteraciones sin mejorar la función objetivo.

Ambos criterios se suministran como parámetro de entrada al algoritmo; en el primero de los criterios de parada, si luego de transcurrido un tiempo preestablecido no se mejora la mejor solución hallada, el algoritmo entrega la solución que mejor se ajusta a los requerimientos del problema de optimización. En el segundo de los criterios de parada, si luego de un número de iteraciones no se mejora la solución hallada, el algoritmo entrega la solución que mejor se ajusta a los requerimientos del problema de optimización.

9.4 RESULTADOS OBTENIDOS

A continuación, se presenta tabla con 100 resultados, donde se registran las pruebas realizadas del algoritmo genético, en la cuales se tienen individuos de tamaño máximo 54 (Genes) a mínimo 14 (Genes).

Tabla 2 Resultados de pruebas de ejecución

| Prueba | Función Objetivo | Longitud del cromosoma | Cantidad de personas programadas | Tiempo total de la ruta (minutos) |
|--------|---------------------|------------------------------|--|---|
| 1 | 773 | 52 | 11 | 477.5 |
| 2 | 691 | 52 | 11 | 470.56 |
| 3 | 570 | 42 | 8 | 475.69 |
| 4 | 511 | 22 | 9 | 477.94 |
| 5 | 407 | 27 | 7 | 471.67 |
| 6 | 448 | 27 | 7 | 456.91 |
| 7 | 639 | 27 | 9 | 470.93 |
| 8 | 599 | 27 | 8 | 472.49 |
| 9 | 523 | 27 | 7 | 472.28 |
| 10 | 531 | 27 | 8 | 472.64 |
| 11 | 562 | 17 | 7 | 451.5 |
| 12 | 607 | 17 | 9 | 472.11 |
| 13 | 627 | 17 | 10 | 475.51 |
| 14 | 358 | 17 | 6 | 462.98 |
| 15 | 724 | 17 | 10 | 472.85 |
| 16 | 540 | 17 | 7 | 475.14 |
| 17 | 366 | 17 | 6 | 465 |
| 18 | 475 | 17 | 8 | 477.61 |
| 19 | 676 | 16 | 10 | 454.57 |
| 20 | 605 | 16 | 11 | 462.52 |
| 21 | 524 | 16 | 9 | 478.94 |
| 22 | 683 | 16 | 9 | 439.71 |
| 23 | 571 | 16 | 8 | 475.67 |
| 24 | 580 | 16 | 8 | 461.68 |
| 25 | 635 | 15 | 9 | 455.7 |
| 26 | 665 | 15 | 9 | 477.98 |
| 27 | 665 | 15 | 9 | 477.98 |
| 28 | 688 | 15 | 9 | 473.64 |
| 29 | 703 | 15 | 9 | 469.94 |
| 30 | 465 | 14 | 8 | 473.71 |
| 31 | 487 | 14 | 9 | 467.53 |
| 32 | 514 | 14 | 8 | 471.19 |
| 33 | 704 | 14 | 12 | 444.23 |
| 34 | 619 | 14 | 10 | 465.48 |
| 35 | 710 | 14 | 10 | 476.96 |
| 36 | 713 | 14 | 11 | 468.95 |
| 37 | 620 | 14 | 9 | 473.38 |
| 38 | 706 | 14 | 11 | 476.57 |
| 39 | 416 | 14 | 7 | 478.73 |
| 40 | 650 | 14 | 10 | 468.52 |
| 41 | 577 | 14 | 7 | 460.34 |
| 42 | 693 | 14 | 9 | 451.64 |
| 43 | 578 | 32 | 8 | 479.66 |
| 44 | 468 | 30 | 8 | 468.35 |
| 45 | 666 | 29 | 9 | 435.1 |
| 46 | 642 | 28 | 9 | 471.63 |
| 47 | 579 | 27 | 8 | 464.46 |
| 48 | 529 | 25 | 7 | 462.03 |

| Prueba | Función Objetivo | Longitud del cromosoma | Cantidad de personas programadas | Tiempo total de la ruta (minutos) |
|--------|---------------------|------------------------------|--|---|
| 49 | 364 | 24 | 5 | 471.58 |
| 50 | 642 | 22 | 8 | 434.71 |
| 51 | 500 | 21 | 7 | 451.89 |
| 52 | 583 | 20 | 8 | 478.62 |
| 53 | 515 | 19 | 7 | 439.02 |
| 54 | 583 | 18 | 9 | 476.45 |
| 55 | 476 | 17 | 8 | 457.21 |
| 56 | 643 | 16 | 10 | 475.74 |
| 57 | 663 | 15 | 11 | 478.5 |
| 58 | 739 | 14 | 11 | 467.2 |
| 59 | 466 | 14 | 7 | 477.93 |
| 60 | 678 | 15 | 10 | 466.7 |
| 61 | 673 | 16 | 9 | 447.13 |
| 62 | 634 | 17 | 10 | 479.9 |
| 63 | 515 | 18 | 7 | 446.99 |
| 64 | 698 | 19 | 9 | 476.66 |
| 65 | 660 | 20 | 8 | 468.02 |
| 66 | 636 | 21 | 8 | 440.04 |
| 67 | 622 | 22 | 9 | 466.51 |
| 68 | 637 | 23 | 9 | 461.35 |
| 69 | 695 | 24 | 10 | 455.71 |
| 70 | 634 | 25 | 8 | 478.8 |
| 71 | 382 | 26 | 6 | 470.63 |
| 72 | 631 | 24 | 9 | 472.12 |
| 73 | 476 | 23 | 7 | 471.68 |
| 74 | 468 | 22 | 7 | 475.76 |
| 75 | 611 | 21 | 9 | 468.13 |
| 76 | 678 | 20 | 9 | 463.28 |
| 77 | 601 | 19 | 8 | 415.9 |
| 78 | 679 | 18 | 10 | 467.75 |
| 79 | 585 | 17 | 9 | 470.03 |
| 80 | 660 | 16 | 9 | 474.15 |
| 81 | 715 | 15 | 10 | 454.99 |
| 82 | 701 | 14 | 12 | 450.6 |
| 83 | 382 | 15 | 6 | 468.68 |
| 84 | 505 | 17 | 8 | 464.2 |
| 85 | 610 | 18 | 9 | 477.65 |
| 86 | 580 | 16 | 9 | 466.38 |
| 87 | 590 | 16 | 11 | 458.52 |
| 88 | 525 | 19 | 7 | 446.02 |
| 89 | 639 | 27 | 9 | 470.93 |
| 90 | 607 | 17 | 9 | 472.11 |
| 91 | 577 | 15 | 7 | 475.67 |
| 92 | 678 | 16 | 9 | 457.64 |
| 93 | 615 | 14 | 11 | 479.95 |
| 94 | 693 | 14 | 9 | 455.64 |
| 95 | 506 | 20 | 7 | 451.89 |
| 96 | 467 | 15 | 8 | 467.93 |

| Prueba | Función Objetivo | Longitud del cromosoma | Cantidad de personas programadas | Tiempo total de la ruta (minutos) |
|--------|------------------|------------------------|----------------------------------|-----------------------------------|
| 97 | 635 | 17 | 9 | 479.96 |
| 98 | 645 | 22 | 7 | 438.04 |
| 99 | 478 | 23 | 7 | 471.78 |
| 100 | 605 | 16 | 9 | 460.03 |

Fuente: Autores

9.4.1 Parámetros de ejecución

Los resultados consignados en la Tabla de Resultados de pruebas de ejecución, fueron obtenidos con la ejecución del algoritmo con los siguientes parámetros.

Tabla 3 Parametros de ejecución para el algoritmo

| No | Descripción | Parámetro | Valor |
|----|---|-----------------------------|--------------|
| 1 | Cantidad de elementos cercanos a k , es decir al individuo seleccionado al azar. | cercanos | % (Variable) |
| 2 | Factor para determinar el mínimo de personas aceptado para visitar. | fmin | 0.3 |
| 3 | Limite de la población Genética | limitePoblacion | 2000 |
| 4 | Criterio de parada que permite determinar cuantas generaciones se ejecuta el algoritmo sin mejorar la Función Objetivo. | max_generaciones_sin_mejora | 20000 |
| 5 | Cantidad máxima de individuos genéticos, no factibles permitidos en la población inicial. | max_infactibles | 10% |
| 6 | Recombinación por el método (PMX) <i>Partially Mapped Crossover</i> . | metodoRecombinacion | pmx |
| 7 | Selección por el método de torneo. | metodoSeleccion | torneo |
| 8 | Nombre del archivo en formato <i>JavaScript Object Notation</i> , para el intercambio de datos, contiene los registros de los PPL con la estructura; <pre>{ "index": numero_entero_positivo, "latitude": numero_real, "longitude": numero_real, "peso": numero_entero_positivo }</pre> | nombre_prueba | 'data.json' |
| 9 | Valor usado para la tasa de Mutación | tasaMutacion | 20% |
| 10 | Valor usado para la tasa de Recombinación | tasaRecombinacion | 30% |
| 11 | Tiempo máximo que puede ser usado en la ruta (Restricción). | tiempoMax | 480 Minutos |
| 12 | Tiempo determinado para la toma de datos en la visita de control a cada PPL. | tiempoVisita | 10 Minutos |

Comentado [JCGM20]: Mejora en la redacción, HECHO

Comentado [ADBB21R20]: ok

| No | Descripción | Parámetro | Valor |
|----|---|-------------------|---------|
| 13 | Velocidad promedio para el desplazamiento del servidor penitenciario. | velocidadPromedio | 40 Km/h |

Fuente: Autores

De los resultados de la tabla de pruebas se puede concluir:

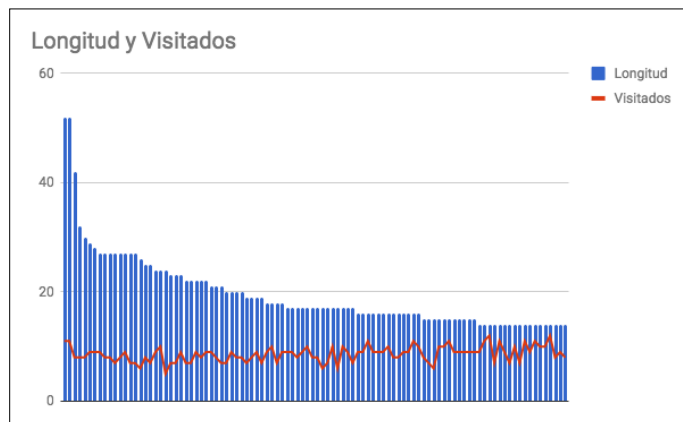
La Función Objetivo de mayor valor es: 773, correspondiente a la iteración a la prueba No 1, donde se logró generar una ruta para 11 elementos visitados en un sujeto de longitud 52 (Genes).

La función objetivo de menos calidad se consiguió en la prueba No 14, donde se logró generar una ruta para 6 elementos visitados en un sujeto de longitud de 17 (Genes).

La función objetivo promedio es de 590,62, el promedio de sujetos visitados es 9, en cuanto al tiempo de visita el promedio en la ruta es de 465 minutos de 480 disponibles.

9.4.2 Gráficas

Gráfica 1 Longitud del Individuo Vs Visitados

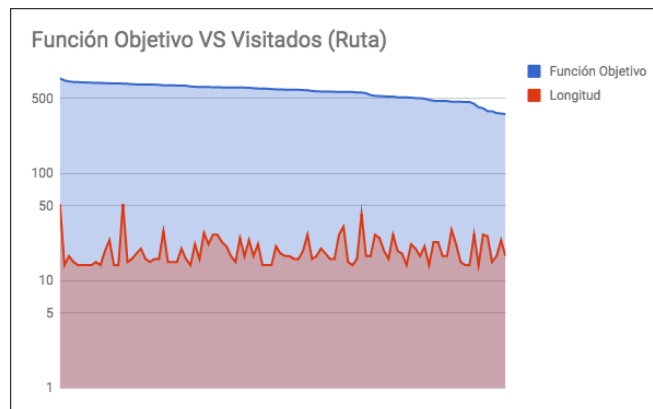


Fuente: Autores

La gráfica anterior permite concluir que no es determinante la longitud del cromosoma vs la cantidad de sujetos visitados en una ruta, por lo anterior se sugiere usar una longitud de 14 (Genes) dado que en el mejor de los casos se puede agregar 12 sujetos a la ruta.

De otra parte al tener cromosomas de menor tamaño permite resolver el problema en menos tiempo y requiere de menos costo computacional.

Gráfica 2 Función objetivo vs Longitud del Individuo



Comentado [JCGM22]: Tenía error, Corregido HECHO

Comentado [ADBB23R22]: ok

Fuente: Autores

En la gráfica anterior se observa que la función objetivo no necesariamente mejora en individuos de mayor longitud.

De la misma manera se puede concluir que el modelo matemático arroja rutas de muy buena calidad, dado que los expertos penitenciarios informan que en el proceso manual que realizan, no superan los cuatro sujetos en promedio por ruta, además que es difícil priorizar de forma objetiva, por la cantidad de posibles soluciones al momento de hacer la planeación.

10. CONCLUSIONES Y TRABAJOS FUTUROS

Ofrecer una solución de automatización para la planeación de vistas a personas privadas de la libertad con medida de aseguramiento en el domicilio, agiliza el proceso de asignación de personas y la ruta a seguir, eliminando la subjetividad que se presenta en el modo manual que está establecido en la actualidad.

Como aporte no esperado en el planteamiento original del problema, se logró además de establecer a cuáles individuos visitar, establecer una ruta de buena calidad a seguir, lo que además permite calcular el costo del combustible del vehículo utilizado para el desplazamiento entre los individuos a visitar.

Se ha logrado establecer un procedimiento organizado alejado de la subjetividad que implica el trabajo manual, encajado dentro del manual de prisión y detención domiciliaria, cumpliendo con los criterios de prioridad de las personas a visitar, y la aleatoriedad requerida.

Se logró verificar la eficiencia de los algoritmos genéticos para ofrecer soluciones de buena calidad, en un problema de optimización específico.

Se aporta a la seguridad ciudadana teniendo en cuenta que, entre mayor rigor en la realización de las visitas a las personas privadas de la libertad con medida de aseguramiento en domicilio, se disminuye la posibilidad de que los individuos burlen los controles para cumplir con las restricciones impuestas por autoridad judicial o aprovechar para delinquir.

Como trabajo futuro se recomienda la implementación del modelo matemático presentado sobre el algoritmo genético modificado de *chu-beasley*, puesto que la selección elitista y la mejoría local que se realiza después de los operadores genéticos de recombinación y mutación, le podrían ofrecer soluciones de mejor calidad; en efecto, se pueden aplicar parámetros de sensibilidad para que algunos elementos no factibles puedan lograr una mejor solución.

11. BIBLIOGRAFÍA

- [1] S. Sabet, F. Farokhi, y M. Shokouhifar, «A novel artificial bee colony algorithm for the knapsack problem», *2012 Int. Symp. Innov. Intell. Syst. Appl.*, n.º July 2012, pp. 1-5, 2012.
- [2] Instituto Nacional Penitenciario y Carcelario INPEC, «MANUAL DE PRISION Y DETENCION DOMICILIARIA». 2010.
- [3] Z. M. David, *How to Solve It: Modern Heuristics*, Second Edi. USA: Springer, 2004.
- [4] D. L. Poole, A. Mackworth, y R. G. Goebel, «Computational Intelligence and Knowledge», *Comput. Intell. A Log. Approach*, n.º Ci, pp. 1-22, 1998.
- [5] S. Norvig, P., & Russell, *Inteligencia artificial*, vol. 1, n.º 3. 2014.
- [6] R. Schank, «Where is the A.I?», *AI Mag.*, vol. 12, n.º 4, p. 38, 1980.
- [7] H. M. Enzensberger y H. E. G. Boeing, «Introducción a la inteligencia artificial», *Behav. Brain Sci.*, pp. 1-12, 1923.
- [8] M. John, «What is Artificial Intelligence?», *Comput. Sci. Dep. Stanford Univ.*, pp. 1-15, 2007.
- [9] N. J. Nilsson, «Logic and artificial intelligence», *Artif. Intell.*, vol. 47, n.º 1-3, pp. 31-56, 1991.
- [10] N. Bizzo y C. El-Hani, «Darwin and Mendel: evolution and genetics», *J. Biol. Educ.*, vol. 43, n.º 3, pp. 108-115, 2009.
- [11] J. H. Holland, «Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence», *October*, n.º 1999, pp. 1999-2000, 1992.
- [12] D. E. Golberg, «Genetic Algorithms in Search Optimization & Machine Learning». Springer, p. 412, 1989.
- [13] S. Mohammad-Azari, O. Bozorg-haddad, y X. Chu, *Studies in Computational Intelligence 720 Advanced Optimization by Nature-Inspired Algorithms*. 2018.
- [14] L. Laguna y M. M. Vega, «Algoritmos Genéticos. Una visión práctica», pp. 29-47, 2009.
- [15] A. Tversky y D. Kahneman, «Judgment under Uncertainty: Heuristics and Biases», *Readings Uncertain Reason.*, vol. 185, n.º 4157, pp. 32-39, 1990.
- [16] T. Cormen, C. Leiserson, R. Rivest, y C. Stein, *Introduction to Algorithms*, vol. 25, n.º 4. 2009.
- [17] J. P. OSMAN y I. H. KELLY, *Meta-Heuristics: Theory & Applications*, vol. Softcover. 1996.

- [18] Ministerio de Justicia y del Derecho Colombia, *Subrogados penales, mecanismos sustitutivos de la pena y vigilancia electrónica en el sistema penal Colombiano*. Bogota, 2014.
- [19] T. J. Barth, *A Primer on Scientific Programming with Python*, Second Edi. New York, 2011.
- [20] M. Gestal, D. Rivero, J. R. J. Rabuñal, J. Dorado, y A. Pazos, *Introducción a los algoritmos genéticos y la programación genética*. 2010.
- [21] A. Hussain, Y. S. Muhammad, M. Nauman Sajid, I. Hussain, A. Mohamd Shoukry, y S. Gani, «Genetic Algorithm for Traveling Salesman Problem : Using Modified Partially-Mapped Crossover Operator», *Comput. Intell. Neurosci.*, vol. 2017, p. 7, 2017.